Възможности за информационно моделиране чрез ERD – основни елементи, отношения, примери – презентация по ИТЗЛД

Съдържание:

1. Увод
2. Понятието ER-модел
3. Понятието ER-диаграма
4. Основни елемени и отношения на ERD
5. Примери
6. Източници

**1. Увод**

След утвърждаването на релационния модел са създадени редица модели на данните, чиято основно цел е била разширяване на възможностите за представяне на предметната област. Всички те представляват всъщност едно разширение на релационния модел в семантично отношение. Така се е оформило понятието **семантичен модел** на БД. Създадените семантични модели могат да се разпределят в следните три групи:

* Разширен релационен модел – включва модели, базирани на n-торни отношения между обектите от предметната област
* Бинарен модел – модели, които представят предметната област посредством отношения между двойки обекти
* **Обектно-ориентиран модел** – към тази група се отнасят онези модели, които разделят отношенията в БД на класове, като се изходжа от типа на тези отношения и от характера и взаимовръзките между обектите, които описват

Най-голямо приложение в теорията и практиката на БД са намерили представителите от третата група. Един от тях е **ER-моделът (модел “същност-връзка”)**, който е получил широко разпространение.

**2. Понятието ER-модел**

ER-моделът има следните по-важни характеристики:

* простота на изграждането на ER-диаграмите, които добре представят предметната област в семантично отношение
* възможност на ER-диаграмите да отразяват не само конкретно (статично) състояние на предметната област, но и поведението на същностите и връзките в определен период от време, т.е ER-моделът може да отрази динамиката на предметната област
* разработени са и се прилагат методи за превръщане на ER-модел в релационен модел, формулирани са правила за представяне на естествения език с ER-диаграми, с което се оптимизират процесите на проектиране на БД и се разширяват възможностите на потребителите

В софтуерното инженерство обикновено се формира ER-модел, който трябва да представи необходимите неща за концептуално моделиране на една система, за да може бизнесът да изпълни по-бърз и ефикасен начин съответните бизнес процеси. Следователно, ER-моделът се е превърнал в абстрактен модел на данни, който дефинира структурата на тези данни. Информацията за тях по този начин може да бъде внедрена в реалната база данни, обикновено в повечето случаи става въпрос за релационна база данни.

ER моделът е резултат от систематичен анализ за определяне и описване на това, което е важно за процесите в дадена област на бизнеса. Той не определя само бизнес процесите, а представя също схемата на бизнес данните в графична форма. Обикновено този модел се представя чрез графично представяне на “кутии” (ентита), които са свързани с връзки (релации). Съответните взаимоотношения между ентитата се изразяват с асоциации и зависимости. ER моделът може да представен и вербално, например една сграда може да бъде разделена на един или повече етажа, но всеки етаж може да се намира само в една сграда.

**3. Понятието ER-диаграма**

**ER-диаграмата (ERD – Entity Relationship Diagram)** е графично средство за абстрактно представяне на отношенията “същност-връзка” в реалния обект, разглеждан като съвкупност от същности с определени връзки между тях. Възможно е да се дефинират различни множества от еднотипни същности E1, E2, .., En, чрез които се представят отделните класове от обекти, както и множествата от еднотипни връзки R1, R2, .., Rm за формализация на релациите (отношенията) между тези класове.

По-просто казано ER-диаграмата се използва за моделиране и проектиране на релационни база данни. Тя отразява обектите, техните свойства и взаимотношенията (връзки) между тях.

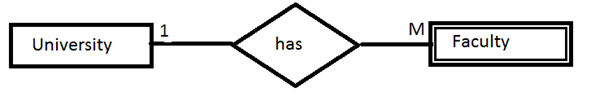
**4. Основни елементи на ERD**

Основните елементи на ERD са обобщени в таблица 1.1. Отношенията могат да бъдат “от много към един” (M:1), “от един към много” (1:M), “от един към един” (1:1) и от “много към много” (M:M).

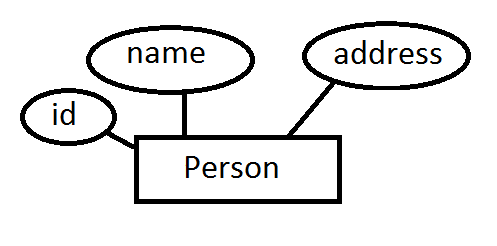
Таблица 1.1

|  |  |
| --- | --- |
| Графично представяне | Описание |
| същност | Обект или концепция, за която ще се съхраняват данни (позволява представяне на клас от обекти) |
| същност | Същност, за която свързаните с нея атрибути се явяват свойства или характеристики (“слаба” същност) |
| атрибут | Основен (ключов) атрибут на дадена същност, който е уникален и съществен за същността |
| атрибут | Многозначен атрибут, който може да приема повече от една стойност (например, даден служител може да притежава няколко умения) |
| атрибут | Атрибут-следствие, който произлиза от друг атрибут (например, месечната заплата на служител е следствие от годишната му заплата) |
| отношение | Отразява връзката между класове от обекти (същности) и определя как те си разпределят информацията в базата данни |

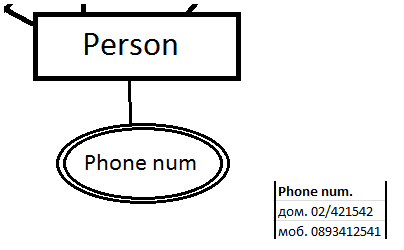
* **Същност (entity)** – представлява всеки един самостоятелен обект от предметната област, например човек, студент, кола, сграда
* **Характеризиращ обект** – обект, който не може да съществува извън рамките на предметната обаст; например “факултет” е характеризиращ обект в една база данни за университети; това е така, тъй като трудно бихме могли да намерим друго значение на думата факултет, от това, което влагаме в нея, когато говорим за университети



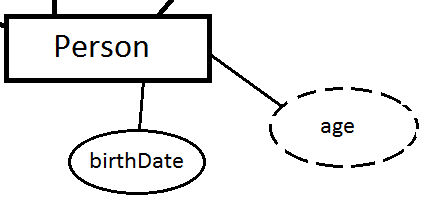
* **Атрибут** – представлява специфичните свойства на обекта



* **Атрибут с множество стойности (Multivalue attribute)** – атрибут, който обикновено е съставен от повеча от една стойност; например телефонен номер – тип на телефона и самият телефон

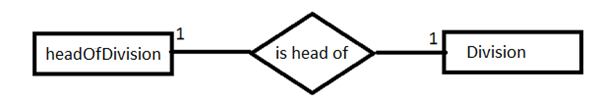


* **Derived attribute** – такъв атрибут, който може да бъде извлечен от други атрибути на същата или на друга релация

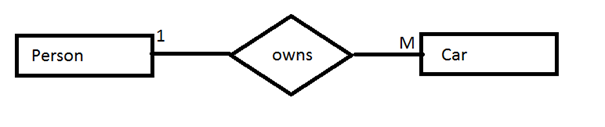


* **Връзки (relationships)** – представят се като ромбове, в средата на които се пише името на връзката, а в двата края до релациите се поставя кардиналността на релацията; кардиналността на връзката представлява число, съответстващо на броя на обектите от едната релация, които ще бъдат асоциирани към друга релация

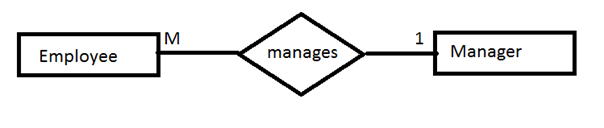
**Връзка 1:1 (One to One)** – един обект от едната релация може да бъде асоцииран с точно един обект от другата и обратно; например един шеф е шеф само на един отдел, а един отдел има точно един шеф от таблицата с шефове на отдели



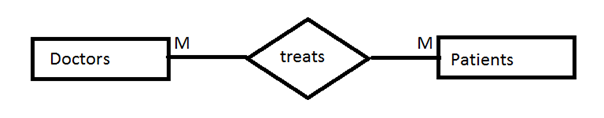
**Връзка 1:M (One to Many)** – един обект от лявата релация може да бъде асоцииран с повече от един обект от дясната релация; например един човек може да притежава много коли, но колата се притежава от един човек



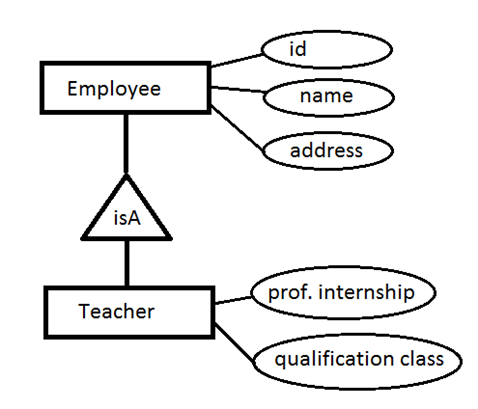
**Връзка M:1 (Many to One)** – повече от един обект от лявата релация могат да бъдат асоциирани с един обект от дясната релация; например един работник се управлява от един мениджър, но мениджърът управлява много работници



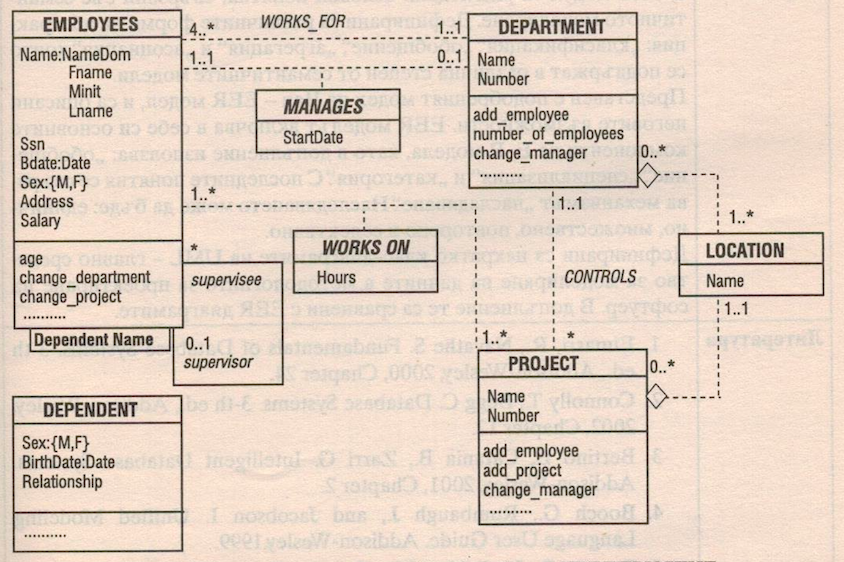
**Връзка M:M (Many to Many)** – повече от един обект от лявата релация се асоциират с повече обекти от дясната релация и обратното; например един лекар може да лекува много пациенти и един пациент може да бъде лекуван от много лекари

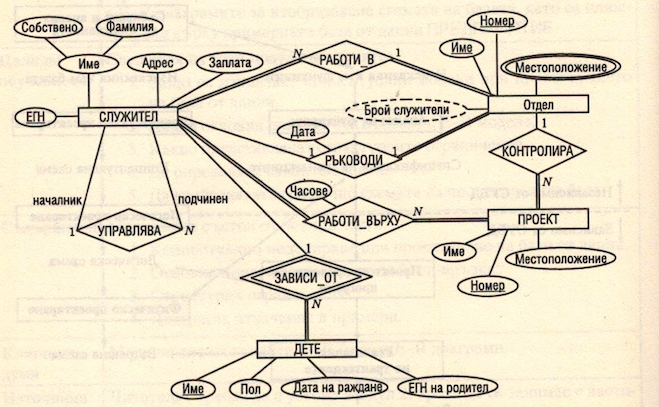


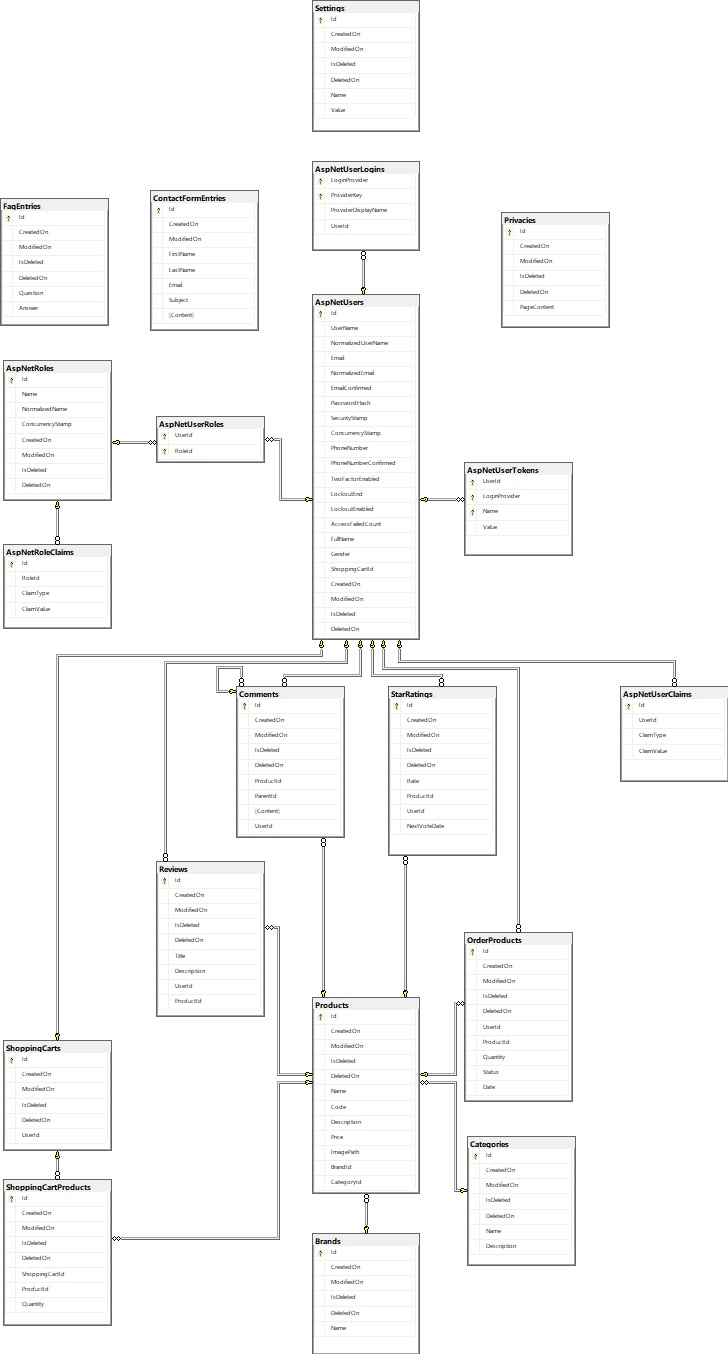
* **Подклас обекти** – това са такива обекти, които притежават всички свойства на даден обект, но и добавят свои допълнителни атрибути; връзката между тях се отбелязва с триъгълник; типичен пример за реализиране на връзка тип “IS-A”; например представете си, че имаме две таблици – таблица с работници и таблица с учители; учителят е вид работник, но освен с традиционните си атрибути – име, адрес и т.н, той има и професионален стаж, както и клас квалификация



**5. Примери**

****

****



6. Източници

* Романски, Р., Нониска, И. Информационни технологии и защита на данните в бизнеса. София, Авангард Прима, 2015
* Знания от предишни курсове по ООП и СУБД
* Лекции на доц. Анна Розева от курса СУБД
* <http://javac.bg>
* <https://www.cphpvb.net>
* Wikipedia